Week 3: Assignment

1. Which of the following statements are true?
   a. The "assign" statement can be used to implement both combinational as well as sequential circuits.
   b. The use of conditional operator in an "assign" statement always results in a multiplexer.
   c. A variable index used at the RHS of an "assign" statement generates a multiplexer whereas usage of it at the LHS results in a decoder.
   d. All of the above.

   Answer: (a) and (c)

   The "assign" statement can be used to describe combinational and sequential statements. Thus option (a) is true. Use of conditional operator in "assign" statements may also generate a latch. Thus option (b) is false. A variable index in the RHS of an "assign" statement always result in a multiplexer whereas in the LHS result in a decoder. Thus option (c) is also true.

2. Which of the following are true for the following code segment?

```
input [7:0] X;
input [0:2] I;
input Y;
output Z;
assign Z = Y ? X[I]: Z;
```

   a. One 8-to-1 and one 2-to-1 multiplexers will be generated.
   b. One 2-to-1 multiplexer will be generated.
   c. One 8-to-1 multiplexer and one d-latch will be generated.
   d. None of the above.

   Answer: (c)

   The use of variable index "I" in the RHS result in an 8-to-1 multiplexer whereas since the output Z will only change when Y is active it will also result in a D-latch. Thus option (c) is true.

3. Which of the following constructs will be generating a decoder / demultiplexer and a multiplexer, where "X", "Y", "Z" and "I" are variables?
   a. assign X = Y[I] & Z ;
   b. assign X[Z] = ~Y ? I : 1'b1 ;
   c. assign Y = ~X ? Z[I] : 1'b0;
   d. assign Z[I] = X & ~Y;

   Answer: (b)

   Option (a) will generate a multiplexer as index variable "I" appears at the RHS of "assign" statement. The usage of index variable "Z" in the LHS of "assign" statement in option (b) will generate a decoder as well as a multiplexer due to the condition at the RHS. Similarly option (c) will generate two multiplexer and option (d) will generate a decoder only. Thus option (b) is the correct answer.

4. What does the following code segment implement, assume the variables I, W, X and Y are of 1-bit size?

```
assign X = ~(Y & W & I);
assign Y = ~(I & Z & X);
```

   a. A combinational circuit.
   b. A 1-bit level triggered latch.
   c. A 1-bit edge triggered latch.
   d. None of the above.

   Answer: (b)

   Here the code segment will generate a 1-bit level triggered latch where W & Z are latch inputs, X & Y are latch outputs and I is the enable input. Thus option (b) is correct answer whereas option (a) and (c) are incorrect.

5. Which of the following are true for procedural blocks?
   a. The "always" block is used for simulation and synthesis both.
   b. The "initial" block can only be used in synthesis.
   c. The "initial" and "always" block both may contain sequential statements.
   d. The "initial" block executes only once whereas "always" block executes infinite times and often associated with an event.

   Answer: (a), (c) and (d)

   The "always" block can be used for both simulation and synthesis whereas "initial" block is only used in test benches. Thus option (a) is true and option (b) is false. The "initial" and "always" blocks both are procedural blocks. Thus option (c) is true. The "initial" block executes once at the beginning of simulation whereas "always" block is like a non terminating loop that may appear with optional event expression. Thus option (d) is also true.

6. What does the following code segment indicate?

```
initial  clk = 1'b0;
always #5 clk = ~clk;
```

   a. Raising edges of the clock will appear at times 5, 10, 15, 20, ...
   b. Raising edges of the clock will appear at times 10, 20, 30, 40, ...
   c. Raising edges of the clock will appear at times 5, 15, 25, 35, ...
   d. None of the above

   Answer: (c)

   Here since the clock that is initialized with logic value 0 toggles after each 5 time unit, the clock period is 10 time unit and raising edge of the clock will appear at times 5, 5+10, 5+10+10, ... and so on. Thus option (c) is correct.

7. Which of the following statements are true about the following code segment:

```
module example(a, b, x, y, clk, set, reset);
input a, b, clk, set, rst;
output reg x, y;
always @(negedge clk, posedge set, posedge
reset)
  begin
     if (set == 1)
       begin x <= 1; y <= 0; end
     else if (reset == 1)
       begin x <= 0; y <= 1; end
     else
       begin
             x <= ~(a & y & ~clk);
             y <= ~(x & ~clk & b);
       end
  end
endmodule
```

a. The module generates a latch that operates on the failing edge of 'clk' signal.
b. The module generates a level sensitive latch operating on the raising edge of the 'clk' signal.
c. The module produces a latch with synchronous active high set and reset.
d. The module produces a latch with asynchronous active high set and reset.

Answer: (a) and (d)

Here the module will generate a 2-input (a, b) latch operating on the negative edge of the 'clk' as described in the event expression. Thus option (a) is true and option (b) is false. The latch can be set or reset independent of 'clk' signal using 'set' or 'reset' signal active high. Thus option (c) is false and option (d) is true.

8. Which of the following is true for the module given below?

```
module mydesign (a, b, c);
input c;
output reg a, b;
always @(c)
begin
     if (c == 1'b0)
        begin b <= ~a; a <= ~(c | b); end
     else if (c == 1'b1) a <= ~(b ^ c);
end
endmodule
```

a. A 2-to-1 multiplexer will be generated.
b. A latch with enable signal c will be generated for the output b.
c. A pure combinational circuit using NOT, NOR and XNOR logic gates will be implemented.
d. The synthesis tool will give an error.

Answer: (a) and (b)

Since variable ‘a’ is assigned to different combinatorial expression based on state of selection variable ‘c’, a 2-to-1 multiplexer will be generated and a latch will also be inferred to retain the value of variable ‘b’ as ‘b’ must retains its value when variable ‘c’ is in logical state ‘1’. Thus option (a) and (b) are true whereas option (c) and (d) are false.

9. Which of the following is true for the following code block?

```
output reg clk1, clk2;
initial
     begin
          clk1 = 1’b 0;
          clk2 = 1’b 1;
          forever clk1 = ~clk1;
          repeat (50) #5 clk2 = ~clk2;
          #75 $finish;
     end
```

a. The state of ‘clk2’ variable will change 50 times.
b. The variable ‘clk1’ will change its state indefinitely until terminates at time 75 units.
c. The variable ‘clk2’ will never change its initial state.
d. The block will never terminate.

Answer: (c) and (d)

Since the “forever” loop is specified without delay, the statement will execute indefinitely without advancing time and “repeat” and “$finish” statement will never be executed. Thus option (a) and (b) are false whereas option (c) and (d) are true.

10. Which of the following is true about the following code segment

```
integer x, y;
initial
     begin
          x = 15;
          y = 10;
     end;
initial
     repeat (x) $display (“X=%d”, x);
initial
     while (y < 12)
          begin y = y + 1; x = x - 1; end
```

a. The simulation will print the current value of ‘x’ 15-times.
b. The simulation will print the current value of ‘x’ 13-times.
c. It cannot be determined exactly how many times the value of ‘x’ will be printed.
d. The simulation will always display 15 as the value of ‘x’.

Answer: (a)

Since the variable ‘x’ used in repeat loop expression will be evaluated only once when the loop starts and initially value of ‘x’ is 15, the current value of ‘x’ will be printed 15 times due to the $display statement used in loop body. Thus option (a) is true whereas option (b) and (c) are false. Option (d) is also false as the value of ‘x’ gets updated in another initial block.